AF

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of

SUTO, AKIO

Docket No: Q60558

Appln. No.: 09/819,612

Group Art Unit: 2161

Confirmation No.: 4173

Examiner: Etienne Pierre LEROUX

Filed: March 29, 2001

For: DISTRIBUTED DATA PROCESSING SYSTEM AND METHOD OF PROCESSING DATA IN DISTRIBUTED DATA PROCESSING SYSTEM

SUBMISSION OF APPEAL BRIEF

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Submitted herewith please find an Appeal Brief. The USPTO is directed and authorized to charge the statutory fee of $500.00 and/or all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account. A duplicate copy of this paper is attached.

Respectfully submitted,

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE
23373
CUSTOMER NUMBER

Susan Perng Pan
Registration No. 41,239

Date: March 1, 2006

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of

SUTO, AKIO

Docket No: Q60558

Appln. No.: 09/819,612

Group Art Unit: 2161

Confirmation No.: 4173

Examiner: Etienne Pierre LEROUX

Filed: March 29, 2001

For: DISTRIBUTED DATA PROCESSING SYSTEM AND METHOD OF PROCESSING DATA IN DISTRIBUTED DATA PROCESSING SYSTEM

APPEAL BRIEF UNDER 37 C.F.R. § 41.37

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 41.37, Appellant submits the following:

Table of Contents

I. REAL PARTY IN INTEREST

The real party in interest in this appeal is Fuji Photo Film, Co., Ltd. of Japan. The assignment was previously submitted and was recorded on March 29, 2001 at Reel 011662, Frame 0420.

II. RELATED APPEALS AND INTERFERENCES

To the knowledge and belief of Appellant, the Assignee, and the Appellant's legal representative, there are no other appeals or interferences before the Board of Appeals and Interferences that will directly affect or be affected by the Board's decision in the instant Appeal.

III. STATUS OF CLAIMS

Claims 1-19 are pending in the present application and stand finally rejected.

Claims 1-2, 7-9 and 16- 19 have been rejected under 35 U.S.C. § 102 as being anticipated by Mayhead (U.S.P. 6,367,029). Claims 3-4, 10-12 and 14 have been rejected under 35 U.S.C. § 103 as being unpatentable over Mayhead in view of Makinen (U.S.P. 5,758,067). Claims 5-6 have been rejected under 35 U.S.C. § 103 as being unpatentable over Mayhead and further in view of Nakamura (U.S.P. 5,347,463). Claims 13 and 15 have been rejected under 35 U.S.C. § 103 as being unpatentable over Mayhead in view of Nguyen (U.S.P. 6,202,070).

A copy of the pending claims on appeal is set forth in an attached Appendix.

IV. STATUS OF AMENDMENTS

A Response Under 37 C.F.R. § 1.116 was filed October 6, 2005 in response to the final Office Action dated June 6, 2005. The response included no claim modifications. All arguments are believed to have been previously entered and made of record.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

Appellant's invention as recited in independent claim 1 relates to a distributed data processing system. The system includes multiple servers (Fig. 1, elements 12, 14) and clients (Fig. 1, elements 20, 22, 24, 26, 30, 32). In an exemplary embodiment, in the event of database memory update to a first server (12, 122) through a client data process, the database memory of a second server (14, 142) also becomes updated through activation of a replication signal (132) and update of the database memory of the second server (14). Each server includes a backup memory (124, 144) which are updated at certain intervals. An archive data memory (126, 146) stores data based on updating information of the database as archive data. Database update information transfer unit (130) transfers updating information of the data (122) based on the replication trigger signal provided by the replication trigger generator (132). Database updating processor (128) updates information of the database as archive data. At least part of the data is updated or recovered using the archive data. This allows the servers to continue operating while maintaining consistency in the data between servers. Pages 8 and 24-29.

Appellant's invention as recited in independent claim 7 includes analogous recitations.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1-2, 7-9 and 16-19 have been rejected under 35 U.S.C. § 102 as being anticipated by Mayhead.

2. Claims 3-4, 10-12 and 14 have been rejected under 35 U.S.C. § 103 as being unpatentable over Mayhead in view of Makinen.

3. Claims 5-6 have been rejected under 35 U.S.C. § 103 as being unpatentable over Mayhead and further in view of Nakamura.

4. Claims 13 and 15 have been rejected under 35 U.S.C. § 103 as being unpatentable over Mayhead in view of Nguyen.

VII. ARGUMENT

The claims should be considered in at least four separate groups, as set forth in Section VI. However, for the claims of Group 1, claim 19 does not stand or fall with the remainder of the claims. Rather claim 19 is separately patentable for the reasons set forth below.

Group 1: Argument 1 Mayhead does not anticipate the independent claims 1 and 7

Mayhead relates to a file server system which is tolerant to software and hardware failures and located over a plurality of hardware nodes. The nodes of the system act as hosts for software components of the system, several of which software components can be replicated. The replicable software components include the system file store, a checker and a logger. The replicated components have one primary copy and one or more back-up copies. Replica copies of a given replicated component are each located at different nodes. Location and handling of replica copies of a given replicable component is under the control of a replication manager which is a (non-replicable) distributed software component of the system.

The replication manager operates in failure recovery and management of nodes entering and leaving the system. See col. 2, lines 29-51. The replication manager also tracks the replicas of components such as the checker or the logger components, rather than individual database items. The database content management is performed by a signature checker. In particular, upon a client request to write data to a first file store, a signature is computed for the data to be written. The data is transmitted along with the signature to a backup store (of the same file store) and the checker. The backup stores the new data. The checker stores subsequently stores the

new signature and a multicast command is sent to indicate that data is updated and the first file store written. Col. 12, lines 40-67.

The Examiner maintains that Mayhead teaches all features of claim 1. Claim 1 describes a replication trigger generator generating a trigger based on updating of a database, and updating the database of another one of the servers based on the replication trigger. The Examiner contends that the replication manager corresponds to the claimed trigger generator. The Examiner is attempting to make a rejection based on the appearance of any "replication" element without due regard to the remaining claim recitations for that element. In particular, the claimed replication element generates a trigger based on updating of a database. By contrast, the replication manager of Mayhead relates to management of operational software components (e.g. a checker or a logger) and management of the system as such components (in nodes) as they enter and exit the system. Therefore, the Examiner's reliance on the mere disclosure of the replication manager, without due regard to its operations with other system elements, renders the anticipation rejection of claim 1 unsupportable.

Appellant would submit that the form of updating as claimed is not inherent in Mayhead. In particular, any data updating occurring in Mayhead is not necessarily in response to a replication trigger signal caused from a database update. Col. 6, lines 25-45 of Mayhead, reproduced at page 13 of the Detailed Action, indicates that a check (of data) is provided on an **intra-server multicast** basis. In view of a multicast (or broadcast), the update does not necessarily implicate a trigger signal generation. In other words, an alternative form of update can occur. Second, as discussed above, the multicast is of the **intra-server** form, which implies

single server communications. By contrast, claim 1 recites updating of information from one server to another.

To the extent that Mayhead includes storage of data from an external source, this does not meet all requirements of the claim. Final Office Action, Detailed Action at page 12. Assuming *arguendo* that an external update source is provided, there is no requirement that the data must come from a different server which is what is described by claim 1. The external source data can simply be supplied through a user input and not from another server. In this regard, the Examiner's reliance on the general storage disk controller is insufficient to make up the deficiencies discussed above regarding server to server transfer of information.

The Examiner refers to the file server system 60 and col. 4, lines 53-58. However, the file server system 60 is the entire system disclosed in Mayhead, not a database updating processor. Moreover, the cited excerpt is silent regarding this feature of the claim. Instead, the excerpt simply refers to a file server system 60 having a replicated system file store with a primary 1 and a back-up 2. Thus, although Mayhead discloses replicating the primary file store 1 with a back-up file store 2, the reference does not disclose updating the database based on information transferred from another server. Since all communications of Mayhead occur on an intra server link, this suggests that Mayhead relates to a fundamentally different system that lacks inter server relationships as claimed. To the extent any updates are provided, this appears to be made on a broadcast base and not on a replication trigger and updates. Therefore, claim 1 is not anticipated by Mayhead.

As an additional point, Appellant submits that the trigger signal is generated as a result of updating of a database, such as upon a writing (updating) of the database, and updating of database information from one server to another. To the extent Mayhead also teaches maintenance of database consistency, the transfer occurs from a write element to first a back up and then the primary intended target. The order of the writing and the transfer appears to be opposite of that claimed because the triggering features differ. Therefore claim 1 is patentable for all the above reasons. Claim 7 is patentable for analogous reasons, and the remaining claims are patentable based on their dependencies.

Claims 2 and 16-18 are not anticipated by Mayhead, at least because of their dependence from claim 1.

Also, claims 8 and 9 are not anticipated by Mayhead, at least because of their dependence from claim 7.

Group 2: Mayhead does not teach the features of claim 19

With further regard to claim 19, this claim describes updating of the database occurs prior to generation of the trigger. The Examiner cites cols. 7-8 of Mayhead to teach this feature. However, the cited portion at col. 7 merely describes software component replication, not data replication from server to server. The cited portion at col. 8 teaches generation of a signature for comparison prior to writing of data, which (at best) is the opposite of the order of operations of claim 19. Therefore, claim 19 is patentable for this additional reason.

Group 3: Claims 3 and 4 are patentable over Mayhead and Makinen

Makinen relates to an automated tape backup system and method for computer systems. The automated tape backup system facilitates the archiving of files from a computer hard disk to a tape for restoration at a later time. A scheduler controls an automated backup mechanism. The scheduler uses system status information available from the computer system to determine, among other things, what type of backup is being performed.

For the rejection over Mayhead in view of Makinen, Appellant submits that claims 3, 4, 10-12 and 14 are allowable over the prior art, at least because of their dependence from claims 1 and 7, respectively, and because Makinen fails to make up for the deficiencies of Mayhead described above.

Group 4: Claims 5 and 6 are patentable over the combination of Mayhead and Nakamura

Nakamura relates to systems and a method for line production management system for use in production lines which conduct the body during painting of automobiles. The system uses an information code for conveying devices on which an item to be processed is placed has the data of the same item to be processed stored as a set in a memory. The information code inputted into a register undergoes tracking corresponding to the conveyance of the item to be processed. Accordingly, by referring to the identification code in the register and the contents of the memory, it is possible to know the data relating to an item to be processed and the position thereof.

Regarding the rejection of claims 5 and 6, Appellant submits that these claims are allowable over the Mayhead/Nakamura combination, at least because of their dependence from claim 1 and because Nakamura does not make up for the above-identified deficiencies of Mayhead.

Group 5: Claims 13 and 15 are patentable over Mayhead and Nguyen

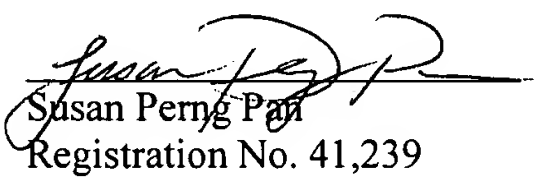
Nguyen relates to a system of software distribution in computer manufacturing which manages and distributes software from release by a software engineering group to installation at a remote manufacturing site or testing facility.

Appellant submits that claims 13 and 15 are allowable over the prior art, at least because of their dependence from claims 1 and 7, respectively, and because Nguyen fails to make up for the deficiencies of Mayhead.

Unless a check is submitted herewith for the fee required under 37 C.F.R. §41.37(a) and 1.17(c), please charge said fee to Deposit Account No. 19-4880.

The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account.

Respectfully submitted,



Susan Perng Pan
Registration No. 41,239

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE
23373
CUSTOMER NUMBER

Date: March 1, 2006

CLAIMS APPENDIX

CLAIMS 1-19 ON APPEAL:

1. A distributed data processing system comprising a plurality of servers and a plurality of clients connected to the servers for performing a distributed data processing process on an object to be controlled,

each of said servers comprising:

a database memory for storing a database which is updated by the distributed data processing process performed by said clients;

a replication trigger generator for generating a replication trigger based on the updating of said database by the distributed data processing process performed by said clients connected to one of the servers;

an updating information transfer unit for transferring updating information of said database to another one of the servers based on said replication trigger;

a database updating processor for updating said database based on the updating information transferred from the other server; and

an archive data memory for storing updating information of said database as archive data;

wherein at least part of said database is recovered using said archive data.

2. A distributed data processing system according to claim 1, wherein each of said clients comprises:

a connection information manager for managing connection information of a connection destination server to which the clients are connected; and

a connection information changer for changing the connection information of the connection destination server;

the arrangement being such that if any of said servers suffers a fault, said connection information is changed by said connection information changer, and the distributed data processing process performed by the clients connected to the server which suffers the fault is continued under the management of another normal one of the servers to which said connection information is changed.

3. A distributed data processing system according to claim 1, further comprising:

a backup processor for performing backup process at predetermined time intervals while said database is in operation;

a backup data memory for storing backup data produced by the backup process performed while said database is in operation;

wherein said database is recovered using said backup data and said archive data.

4. A distributed data processing system according to claim 2, further comprising:

a backup processor for performing backup process at predetermined time intervals while said database is in operation;

a backup data memory for storing backup data produced by the backup process performed while said database is in operation;

wherein said database is recovered using said backup data and said archive data.

5. A distributed data processing system according to claim 1, wherein said servers comprise:

a server for managing one of the clients which is of a production management system which is of the object to be controlled; and

a server for managing one of the clients which is of a process control system which is of the object to be controlled.

6. A distributed data processing system according to claim 5, wherein each of said servers has independent settings of distributed data processing so that said database can be independently processed in inserting, updating, or deleting data.

7. A method of processing data in a distributed data processing system having a plurality of servers and a plurality of clients connected to the servers for performing a distributed data processing process on an object to be controlled, comprising the steps of:

updating a database according to the distributed data processing process performed by said clients;

generating a replication trigger based on the updating of said database by the distributed data processing process performed by said clients connected to one of the servers;

transferring updating information of said database to another one of the servers based on said replication trigger;

updating said database based on the updating information transferred from the other server; and

storing the updating information of said database as archive data;

wherein at least part of said database is recovered using said archive data.

8. A method according to claim 7, further comprising the steps of:

if any of said servers suffers a fault, changing a connection destination of the clients connected to the server which suffers the fault to another normal one of the servers; and

continuing the distributed data processing process performed by the clients connected to the server under the management of the other normal server.

9. A method according to claim 8, further comprising the step of activating again said server suffering the fault to resume normal operation after completion of a restoring process, said step of activating again said server comprising the steps of:

shutting off all the clients connected to said server;

setting again information of the connection destination of the clients;

connecting the clients to said server according to the set information; and

resuming the distributed data processing process in a normal connection state.

10. A method according to claim 7, further comprising the steps of:

performing a backup process at predetermined time intervals while said database is in operation and saving backup data produced by the backup process performed;

generating and saving archive data based on the updating information of the database which is generated after the backup process performed while said database is in operation has started; and

if one of said servers suffers a fault, copying said backup data of another normal one of the servers, and recovering the database from said archive data of the other normal server.

11. A method according to claim 10, further comprising the step of:

copying said backup data while the clients are being continuously operated by said other normal server.

12. A method according to claim 8, further comprising the steps of:

performing a backup process at predetermined time intervals while said database is in operation and saving backup data produced by the backup process performed;

generating and saving archive data based on the updating information of the database which is generated after the backup process performed while said database is in operation has started; and

if one of said servers suffers a fault, copying said backup data of another normal one of the servers, and recovering the database from said archive data of the other normal server.

13. A distributed data processing system according to claim 1, wherein the replication trigger generator converts an updating SQL into a propagating SQL.

14. A distributed data processing system according to claim 3, wherein a file copy operation is used to store the backup data in the backup data memory.

15. A method according to claim 7, further comprising converting an updating SQL into a propagating SQL.

16. The distributed data processing system according to claim 1, wherein the clients connected to any one of said servers are different from the clients connected to another one of said servers.

17. The distributed data processing system according to claim 1, wherein said database updating processor in each of said servers updates said database based on the updating information, the updating information is generated upon an updating request from one of said clients connected to said server, and said database updating processor transfers the updating information of said database to another one of said servers.

18. The distributed data processing system according to claim 17, wherein said database updating processor determines whether the updating information is generated by said server or is transferred from the other one of said servers, and wherein when it is determined that the updating information is transferred from the other one of said servers, a replication trigger generation inhibition is issued to said replication trigger generator.

19. A distributed data processing system according to claim 1, wherein the updating of the database occurs prior to the generating of the replication trigger.

EVIDENCE APPENDIX:

Pursuant to 37 C.F.R. § 41.37(c)(1)(ix), submitted herewith are copies of any evidence submitted pursuant to 37 C.F.R. §§ 1.130, 1.131, or 1.132 or any other evidence entered by the Examiner and relied upon by Appellant in the appeal.

NONE

RELATED PROCEEDINGS APPENDIX

Submitted herewith are copies of decisions rendered by a court or the Board in any proceeding identified about in Section II pursuant to 37 C.F.R. § 41.37(c)(1)(ii).

NONE